Progress on Designing On-board GRB Trigger

David Wren, Jerry Bonnell, Jay Norris

Recap from DC1…

- We used the raw ~ 15 Hz data, decimated only by removal of events with $\{\phi, \theta\} = \{0, 0\} \rightarrow 12$ Hz. This expedient gave us a quasi-realistic background rate against which to search for GRBs.
- We used only one N-event sliding window as the first bootstrap step in searching for significant temporal-spatial clustering. Compute Log {Joint (spatial*temporal) likelihood} for tightest cluster in window:

$$\text{Log}(P) = \Sigma \text{Log}\{ [1 - \cos(d_i)] / 2 \} + \Sigma \text{Log}\{ 1 - (1 + X_i) \exp(-X_i) \}$$

- Log(P) threshold set and used to search for GRB trigger

GLAST

Day 1 Triggers, Panels Top→Bottom:

Log Prob [$\Delta\rho$'s] + Log Prob [Δt's]

Log Prob [$\Delta\rho$'s]

Log Prob [Δt's]

Raw Rate (includes non-recon'ed γ's — but we don't use them!)

Similar approach to previous studies:

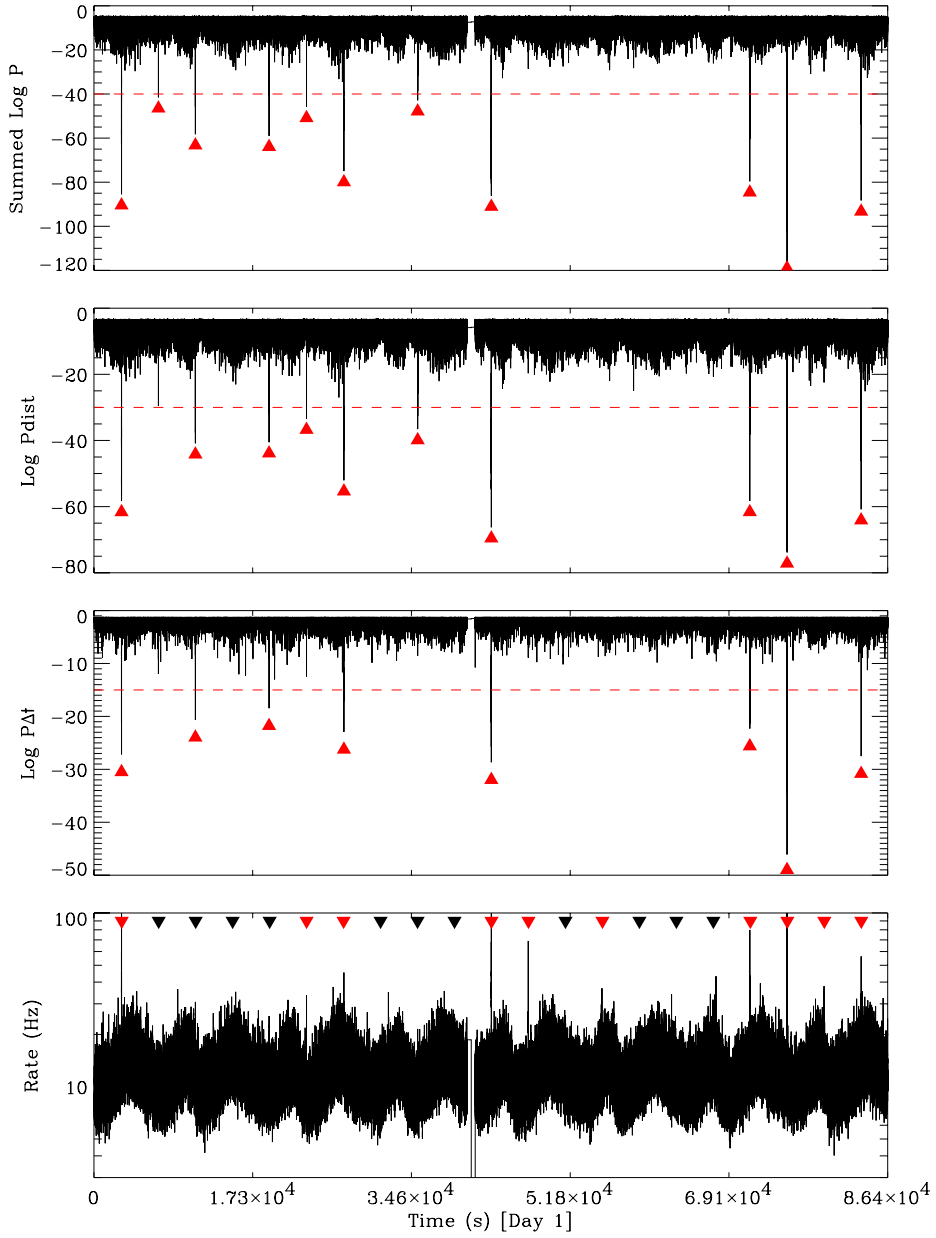
(1) Operate sliding 20-event window;

(2) Find tightest spatial cluster;

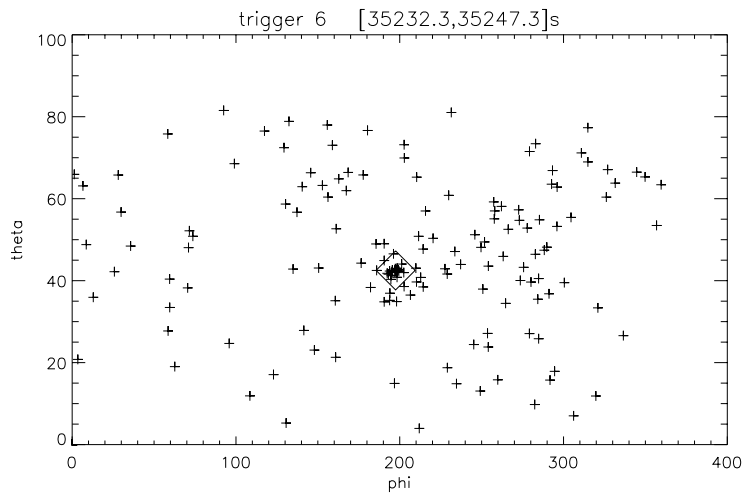
(3) Compute log probs for Δt's, $\Delta\rho$'s in the selected cluster;

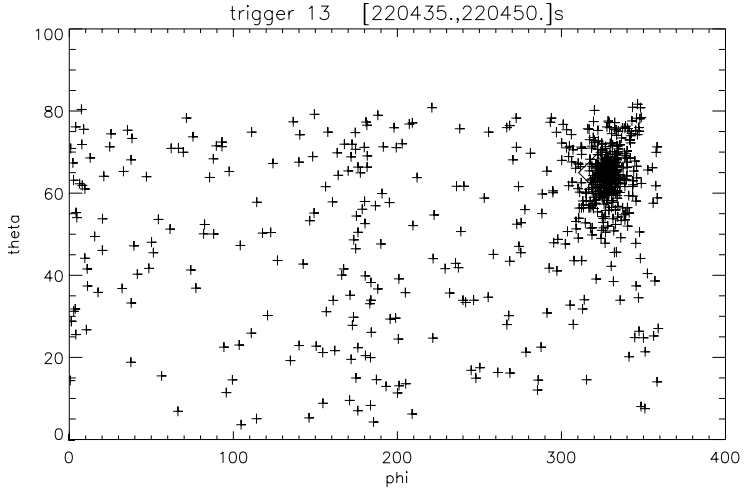
(4) Exceed threshold value, set to allow < 1 false trigger/6 days?

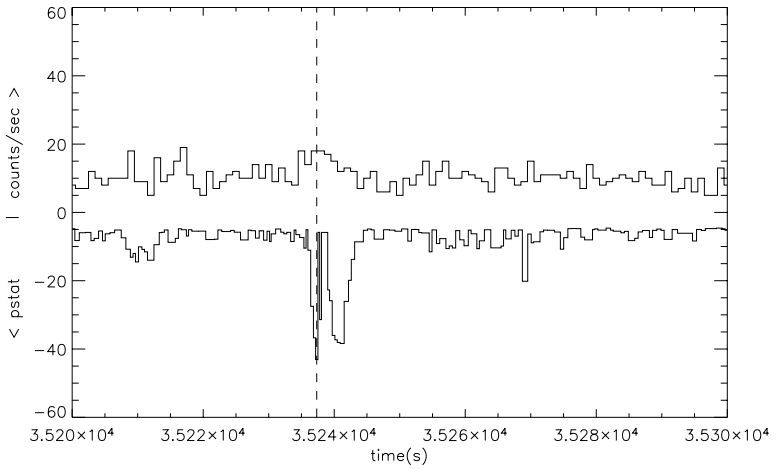
Real Question is: How many "life-like" GRBs would be detected ?

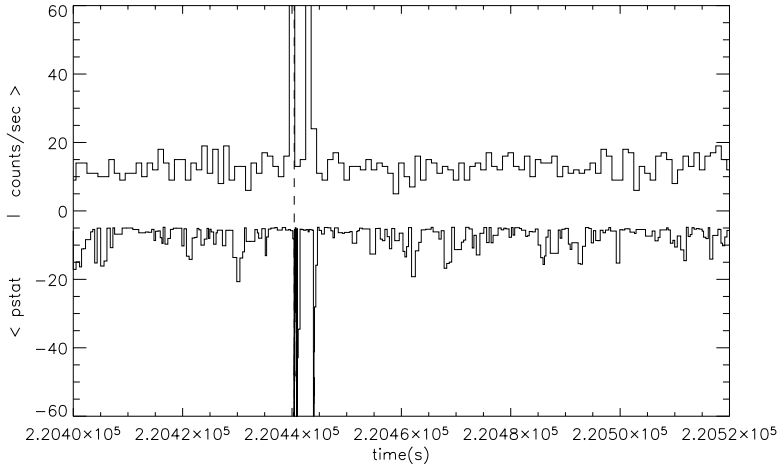


GRB Trigger regions, 2









Nature of On-board GRB Trigger "Task"

- The problem is two-fold:
  - The on-board event trigger rate is expected to be ~ 300 Hz, considerably higher than ~ 30-60 Hz rate for which the unbinned GRB trigger algorithm has been demonstrated to work well …
  - … And the event tracks are often not well enough reconstructed, even at high energy where the nearly straight tracks yield the most accurate contributions to the on-board GRB localization.

How to address:

- 1st, we need to reduce the event rate that will be considered by a "GRB event buffer" — so that more CPU time is available per event for reconstruction computation.

2nd, we need to make fuller use of the on-board hit information, thereby arrive at better reconstruction for events.

General Approach

1. Reduce event rate considered for GRB trigger buffer from ~ 300 Hz to ~ 30-60 Hz: Apply trial subsets of On-board Filters (which are not currently employed), designed to retain as many γ's as possible (what shall be E dependence?) — David Wren has trial filter subsets.

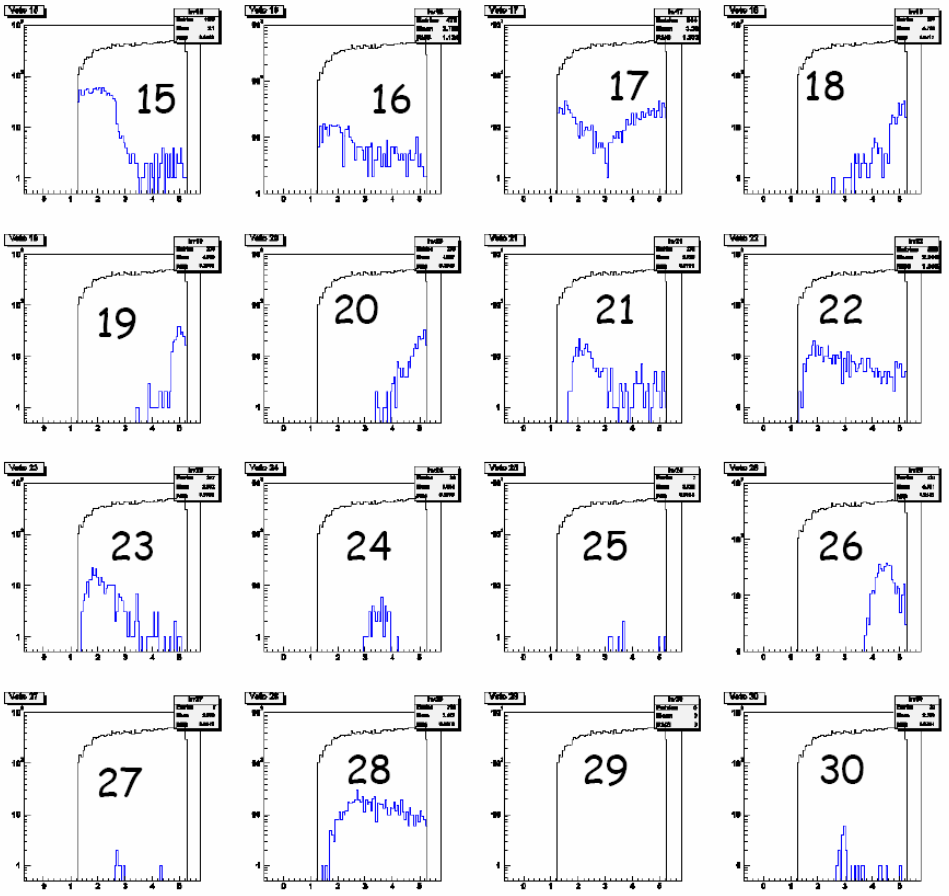
2. Per kept event, utilize hit information across towers, reconstructing longer and more accurate tracks, selecting "best" multi-tower track.

> Promising approach being explored by Wren is "Hough Transform," which picks out colinear hits in polar coordinate space — looks especially good at higher energies where tracks are straighter (tolerance for track curvature might need to be energy-dependent).

3. Execute GRB trigger algorithm, compute Log {Joint (spatial*temporal) likelihood} for tightest cluster in each sliding temporal window:

$$\text{Log}(P) = \Sigma \text{Log}\{ [1 - \cos(d_i)] / 2 \} + \Sigma \text{Log}\{ 1 - (1 + X_i) \exp(-X_i) \}$$

Which Vetoes Killed Gammas?



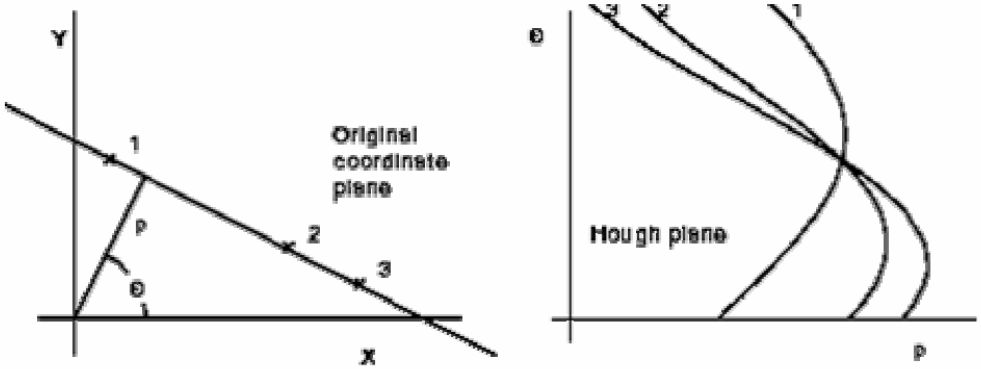
For TKR triggered gammas, there are many active vetoes

Black: TKR triggered events

Blue: TKR triggered and VETOED

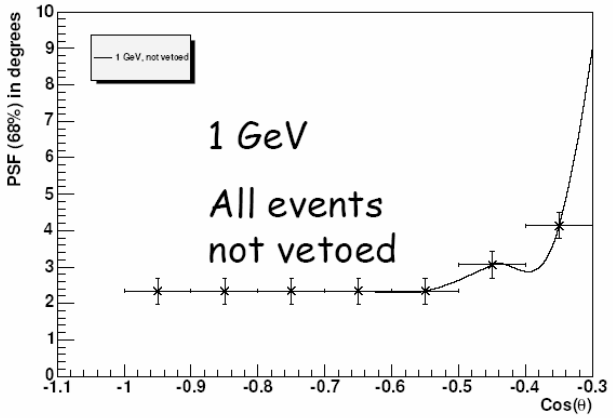
Hough Transform

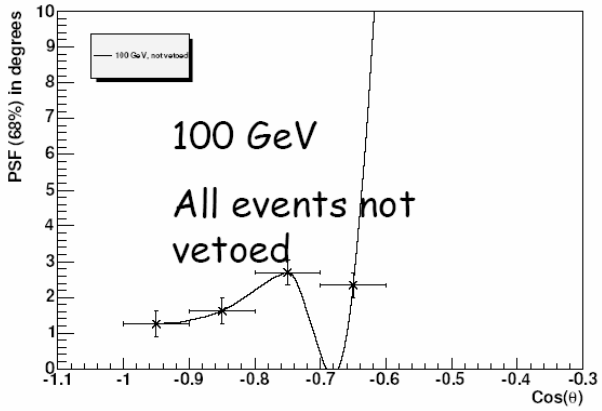
- The basic idea:
  - Transform a coordinate in Cartesian space into a line in (rho, theta) parameter space.
  - Collinear points in Cartesian space have intersecting lines in parameter space
  - Pick out the point of intersection in parameter space, and one gets the line (track) in Cartesian space

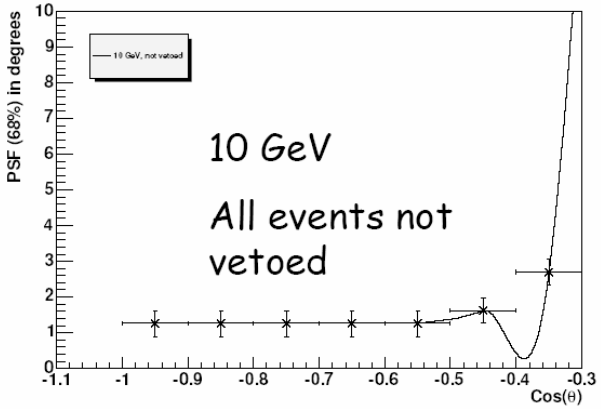


David Wren

Hough Transform: Preliminary Results for All Fvents not Vetoed







Hough starts to perform well around 1 GeV, but still has trouble at extreme angles at 100 GeV

David Wren 25

Upwards Moving Albedo at Lower Energies

- Complicating problem: Upwards moving albedo and low-energy cosmic γ's look very similar, and of course we want to retain the cosmic γ's for the temporal part of likelihood trigger algorithm.

  (Note: Most low-energy γ's will be retained by the standard filters and sent for ground analysis within the ~ 300 Hz telemetry rate.)

- The upwards moving albedo is ~ 140 Hz — alone it is considerably higher than our desired rate (30-60 Hz) to-be-presented to trigger algorithm. Must address this additional background component …

- David Wren has proposed two fairly successful methods for reducing the upwards moving albedo rate, to ~ 60 Hz, applying the methods only for events in outer towers AND with CalEnergySum = 0:

  - Method 1: require start of "3-in-a-row" to be an {X,Y} pair in the same layer, and be the uppermost hits of a track; OR

  - Method 2: if event comes from direction consistent with cone defined by Earth, then kill it.